

2020 자율주행 교육

WeGo 위고 주식회사

1. Yolo ROS

01

Yolo ROS

01. Yolo ROS

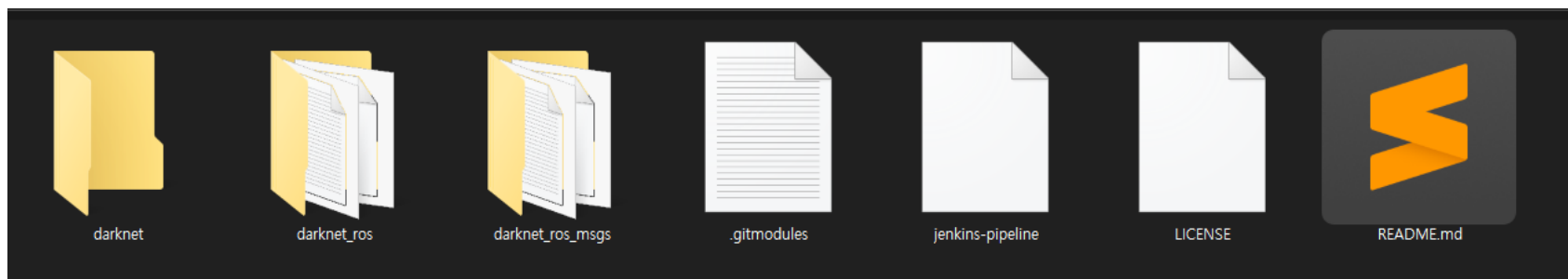
- Yolo와 ROS의 접목
- https://github.com/leggedrobotics/darknet_ros
- `cd catkin_workspace/src`
- `git clone https://github.com/leggedrobotics/darknet_ros`
- `cd ..`
- `catkin_make -DCMAKE_BUILD_TYPE=Release`
- 자동적으로 CUDA 설치 여부 및 OpenCV 확인 후, 맞춰서 Make 진행

01. Yolo ROS

- Training된 Weight파일 다운로드
- `cd catkin_workspace/src/darknet_ros/darknet_ros/yolo_network_config/weights/`
- `wget https://pjreddie.com/media/files/yolov2.weights`
- `wget https://pjreddie.com/media/files/yolov2-tiny.weights`
- Catkin_make 진행 시, 위의 Weight파일이 없을 경우, 자동적으로 Weight파일 다운로드 진행
- 이 후, `roslaunch usb-cam usb-cam-test.launch` (카메라 실행)
- `Roslaunch darknet_ros darknet_ros.launch` 실행

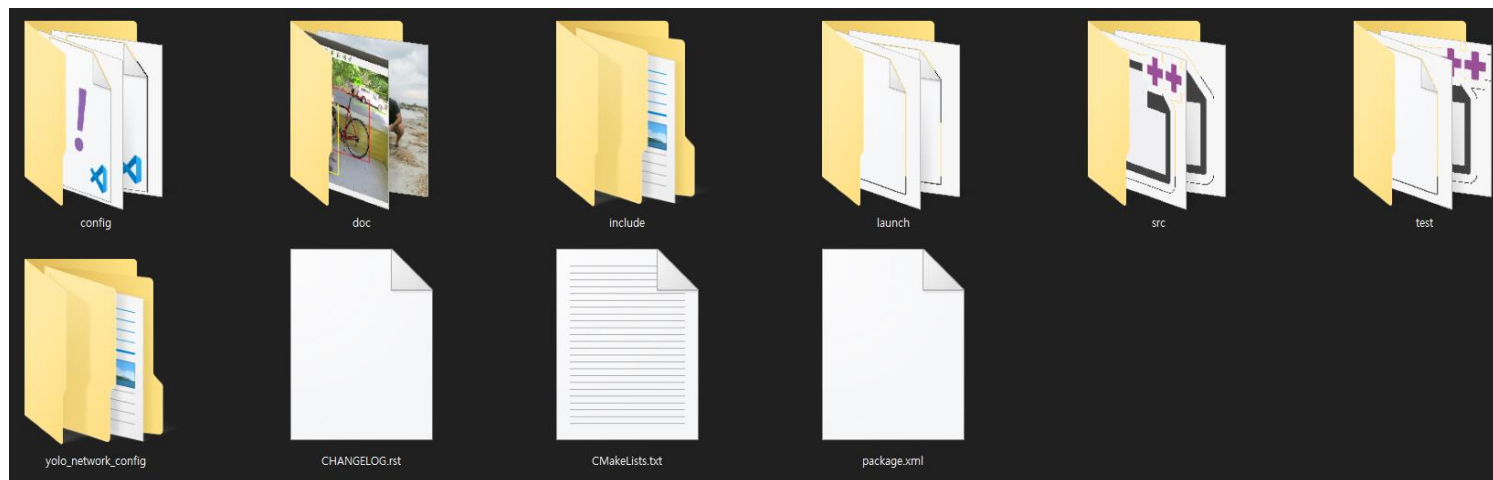
01. Yolo ROS

- 내부 구조 확인
- Darknet = 기존의 Darknet 소스코드가 있는 폴더
- Darknet_ros = catkin_make해서 사용할 darknet 관련 ros 파일이 있는 폴더
- Darknet_ros_msgs = darknet_ros에서 사용할 Topic 및 Action 메시지가 있는 폴더



01. Yolo ROS

- Darknet_ros 폴더 내부 구조 확인
- Config = ros node 실행 시, 사용하는 Weight 및 cfg파일 정보와 ros node 정보가 있는 폴더
- Launch = roslaunch 파일이 모여있는 폴더
- yolo_network_config = darknet_ros에서 사용할 cfg 파일 및 weight 파일이 모여있는 폴더



01. Yolo ROS

- 특별히, CMakeLists.txt 파일과 package.xml 파일을 수정할 필요는 없음
- 사용 시에 수정할 파일은 launch폴더의 darknet_ros.launch 파일과 config 폴더의 ros.yaml 및 기타 weight에 관련된 yaml 파일을 수정하여 사용

01. Yolo ROS

- darknet_ros.launch 파일 살펴보기

```
1 <?xml version="1.0" encoding="utf-8"?>
2
3 <launch>
4   <!-- Console launch prefix -->
5   <arg name="launch_prefix" default=""/>
6   <arg name="image" default="/camera/rgb/image_raw" />
7
8   <!-- Config and weights folder. -->
9   <arg name="yolo_weights_path" default="$(find darknet_ros)/yolo_network_config/weights"/>
10  <arg name="yolo_config_path" default="$(find darknet_ros)/yolo_network_config/cfg"/>
11
12  <!-- ROS and network parameter files -->
13  <arg name="ros_param_file" default="$(find darknet_ros)/config/ros.yaml"/>
14  <arg name="network_param_file" default="$(find darknet_ros)/config/yolov2-tiny.yaml"/>
15
16  <!-- Load parameters -->
17  <rosparam command="load" ns="darknet_ros" file="$(arg ros_param_file)"/>
18  <rosparam command="load" ns="darknet_ros" file="$(arg network_param_file)"/>
19
20  <!-- Start darknet and ros wrapper -->
21  <node pkg="darknet_ros" type="darknet_ros" name="darknet_ros" output="screen" launch-prefix="$(arg launch_prefix)">
22    <param name="weights_path" value="$(arg yolo_weights_path)" />
23    <param name="config_path" value="$(arg yolo_config_path)" />
24    <remap from="camera/rgb/image_raw" to="$(arg image)" />
25  </node>
26
27  <!--<node name="republish" type="republish" pkg="image_transport" output="screen" args="compressed in:=/front_camera/image_raw raw
28  out:=/camera/image_raw" /> -->
29 </launch>
```

01. Yolo ROS

- 사용 시에, Yolo 알고리즘을 적용할 Topic의 이름을 6번 라인의 수정을 통해 사용

```
1 <?xml version="1.0" encoding="utf-8"?>
2
3 <launch>
4   <!-- Console launch prefix -->
5   <arg name="launch_prefix" default=""/>
6   <arg name="image" default="/camera/rgb/image_raw" />
7
8   <!-- Config and weights folder. -->
9   <arg name="yolo_weights_path" default="$(find darknet_ros)/yolo_network_config/weights"/>
10  <arg name="yolo_config_path" default="$(find darknet_ros)/yolo_network_config/cfg"/>
11
12  <!-- ROS and network parameter files -->
13  <arg name="ros_param_file" default="$(find darknet_ros)/config/ros.yaml"/>
14  <arg name="network_param_file" default="$(find darknet_ros)/config/yolov2-tiny.yaml"/>
15
16  <!-- Load parameters -->
17  <rosparam command="load" ns="darknet_ros" file="$(arg ros_param_file)"/>
18  <rosparam command="load" ns="darknet_ros" file="$(arg network_param_file)"/>
19
20  <!-- Start darknet and ros wrapper -->
21  <node pkg="darknet_ros" type="darknet_ros" name="darknet_ros" output="screen" launch-prefix="$(arg launch_prefix)">
22    <param name="weights_path" value="$(arg yolo_weights_path)" />
23    <param name="config_path" value="$(arg yolo_config_path)" />
24    <remap from="camera/rgb/image_raw" to="$(arg image)" />
25  </node>
26
27  <!--<node name="republish" type="republish" pkg="image_transport" output="screen" args="compressed in:=/front_camera/image_raw raw
28 out:=/camera/image_raw" /> -->
29 </launch>
```

01. Yolo ROS

- 14번 라인의 수정을 통해, 사용할 weight와 cfg 파일의 수정이 가능

```
1 <?xml version="1.0" encoding="utf-8"?>
2
3 <launch>
4   <!-- Console launch prefix -->
5   <arg name="launch_prefix" default=""/>
6   <arg name="image" default="/camera/rgb/image_raw" />
7
8   <!-- Config and weights folder. -->
9   <arg name="yolo_weights_path" default="$(find darknet_ros)/yolo_network_config/weights"/>
10  <arg name="yolo_config_path" default="$(find darknet_ros)/yolo_network_config/cfg"/>
11
12  <!-- ROS and network parameter files -->
13  <arg name="ros_param_file" default="$(find darknet_ros)/config/ros.yaml"/>
14  <arg name="network_param_file" default="$(find darknet_ros)/config/yolov2-tiny.yaml"/>
15
16  <!-- Load parameters -->
17  <rosparam command="load" ns="darknet_ros" file="$(arg ros_param_file)"/>
18  <rosparam command="load" ns="darknet_ros" file="$(arg network_param_file)"/>
19
20  <!-- Start darknet and ros wrapper -->
21  <node pkg="darknet_ros" type="darknet_ros" name="darknet_ros" output="screen" launch-prefix="$(arg launch_prefix)">
22    <param name="weights_path" value="$(arg yolo_weights_path)" />
23    <param name="config_path" value="$(arg yolo_config_path)" />
24    <remap from="camera/rgb/image_raw" to="$(arg image)" />
25  </node>
26
27  <!--<node name="republish" type="republish" pkg="image_transport" output="screen" args="compressed in:=/front_camera/image_raw raw
28  out:=/camera/image_raw" /> -->
29 </launch>
```

01. Yolo ROS

- ros.yaml 파일 살펴보기

```
1 ▾ subscribers:
2
3 ▾ camera_reading:
4   topic: /camera/rgb/image_raw
5   queue_size: 1
6
7 ▾ actions:
8
9   camera_reading:
10    name: /darknet_ros/check_for_objects
11
12 ▾ publishers:
13
14 ▾ object_detector:
15   topic: /darknet_ros/found_object
16   queue_size: 1
17   latch: false
18
19 ▾ bounding_boxes:
20   topic: /darknet_ros/bounding_boxes
21   queue_size: 1
22   latch: false
23
24 ▾ detection_image:
25   topic: /darknet_ros/detection_image
26   queue_size: 1
27   latch: true
28
29 ▾ image_view:
30
31   enable_opencv: true
32   wait_key_delay: 1
33   enable_console_output: true
34
```

01. Yolo ROS

- 특별한 수정은 없고, 맨 아래의 image_view의 값을 조정을 통해, 출력을 조정 가능
- enable_opencv값의 수정을 통해, 출력 창을 통해 결과 확인(true) or rviz를 통해 subscrib하여 결과 확인 가능(false)
- enable_console_output값의 수정을 통해, 결과를 Terminal 창에 출력할지 여부를 선택 가능

```
28
29 ▼ image_view:
30
31     enable_opencv: true
32     wait_key_delay: 1
33     enable_console_output: true
34
```

01. Yolo ROS

- yolov2-tiny.yaml 파일 살펴보기

```
1 yolo_model:
2
3   config_file:
4     name: yolov2-tiny.cfg
5   weight_file:
6     name: yolov2-tiny.weights
7   threshold:
8     value: 0.3
9   detection_classes:
10  names:
11    - person
12    - bicycle
13    - car
14    - motorbike
15    - aeroplane
16    - bus
17    - train
18    - truck
19    - boat
20    - traffic light
```

01. Yolo ROS

- 사용할 cfg 파일과 해당하는 weights 파일을 입력하여 사용 가능하며
- 검출 시, 최종적으로 출력될 물체의 최소 Probability를 threshold-value의 수정을 통해 수정 가능(Default는 30% 이상이면 출력)

```
1 ▼ yolo_model:
2
3     config_file:
4         name: yolov2-tiny.cfg
5     weight_file:
6         name: yolov2-tiny.weights
7     threshold:
8         value: 0.3
9 ▼ detection_classes:
10 ▼     names:
11         - person
12         - bicycle
13         - car
14         - motorbike
```

